CLAIMS

We claim:

1. A method of identifying an interface of an application program comprising the interface and source code, said method comprising the steps:

parsing the source code of the application program to identify meta information;

storing in a repository the meta information and a link pointing to an original location of the meta information within the application program;

allowing a user to query the repository to determine which source files and which interfaces comprise the application program;

constructing a new source file containing the interfaces which comprise the application program;

storing the new source file and a link pointing to a location of the new source file in the repository; and

constructing a meta language document containing a description of the application program interfaces to enable a connector building tool to build an interface to the application program.

2. The method of claim 1 further comprising the step of:

for a transaction contained in the application program, displaying a visual indicia which navigates via the link to an entry point of the source code corresponding to transaction.

3. The method of claim 2 further comprising the step of:

displaying the transaction contained in the application program together with a visual indicia which navigates to documentation stored in the repository corresponding to the transaction.

4. The method of claim 3 further comprising the steps of:

allowing the user to select the transaction;

displaying a data structure corresponding to the selected transaction;

allowing the user to select the data structure; and

for the selected data structure, computing input and output fields and overlaying the fields on the selected data structure.

5. The method of claim 4 further comprising the steps of:

allowing the user to edit the computed fields; and

analyzing the selections and editions to determine if an error exists.

6. A method of identifying an interface of an application program comprising the interface and source code, said method comprising the steps:

parsing the source code of the application program to identify meta information;

storing in a repository the meta information and a link pointing to an original location of the meta information within the application program;

allowing a user to query the repository to determine which source files and which interfaces comprise the application program;

for a transaction contained in the application program, displaying a visual indicia which navigates via the link to an entry point of the source code corresponding to transaction;

displaying the transaction contained in the application program together with a visual indicia which navigates to documentation stored in the repository corresponding to the transaction;

allowing the user to select the transaction;

displaying a data structure corresponding to the selected transaction;

allowing the user to select the data structure; and

for the selected data structure, computing input and output fields and overlaying the fields on the selected data structure;

allowing the user to edit the computed fields;

analyzing the selections and editions to determine if an error exists.

constructing a new source file containing the interfaces which comprise the application program;

storing the new source file and a link pointing to a location of the new source file in the repository; and

constructing a meta language document containing a description of the application program interfaces to enable a a connector building tool to build an interface to the application program.

7. An article of manufacture for use in a computer system for identifying an interface of an application program comprising the interface and source code, said article of manufacture comprising a computer-readable storage medium having a computer program embodied in said medium which causes the computer system to execute a method comprising the steps of:

parsing the source code of the application program to identify meta information;

storing in a repository the meta information and a link pointing to an original location of the meta information within the application program;

allowing a user to query the repository to determine which source files and which interfaces comprise the application program;

constructing a new source file containing the interfaces which comprise the application program;

storing the new source file and a link pointing to a location of the new source file in the repository; and

constructing a meta language document containing a description of the application program interfaces to enable a connector building tool to build an interface to the application program.

8. The article of manufacture of claim 7 wherein the embodied computer program further causes the computer system to execute the method step:

for a transaction contained in the application program, displaying a visual indicia which navigates via the link to an entry point of the source code corresponding to transaction.

9. The article of manufacture of claim 8 wherein the embodied computer program further causes the computer system to execute the method step:

displaying the transaction contained in the application program together with a visual indicia which navigates to documentation stored in the repository corresponding to the transaction.

10. The article of manufacture of claim 9 wherein the embodied computer program further causes the computer system to execute the method steps:

allowing the user to select the transaction;

displaying a data structure corresponding to the selected transaction;

allowing the user to select the data structure; and

for the selected data structure, computing input and output fields and overlaying the fields on the selected data structure.

11. The article of manufacture of claim 10 wherein the embodied computer program further causes the computer system to execute the method steps:

allowing the user to edit the computed fields; and

analyzing the selections and editions to determine if an error exists.

12. A article of manufacture of identifying an interface of an application program comprising the interface and source code, said article of manufacture comprising a computer-readable storage medium having a computer program embodied in said medium which causes the computer system to execute a method comprising the steps of:

parsing the source code of the application program to identify meta information;

storing in a repository the meta information and a link pointing to an original location of the meta information within the application program;

allowing a user to query the repository to determine which source files and which interfaces comprise the application program;

for a transaction contained in the application program, displaying a visual indicia which navigates via the link to an entry point of the source code corresponding to transaction;

displaying the transaction contained in the application program together with a visual indicia which navigates to documentation stored in the repository corresponding to the transaction;

allowing the user to select the transaction;

displaying a data structure corresponding to the selected transaction;

allowing the user to select the data structure; and

for the selected data structure, computing input and output fields and overlaying the fields on the selected data structure;

allowing the user to edit the computed fields;

analyzing the selections and editions to determine if an error exists.

constructing a new source file containing the interfaces which comprise the application program;

storing the new source file and a link pointing to a location of the new source file in the repository; and

constructing a meta language document containing a description of the application program interfaces to enable a a connector building tool to build an interface to the application program.

13. A computer system for identifying an interface of an application program comprising the interface and source code, said computer system comprising:

a parser for parsing the source code of the application program to identify meta information;

storage for storing in a repository the meta information and a link pointing to an original location of the meta information within the application program;

a query the repository to determine which source files and which interfaces comprise the application program;

a new source file containing the interfaces which comprise the application program;

storage for storing the new source file and a link pointing to a location of the new source file in the repository; and

a meta language document containing a description of the application program interfaces to enable a a connector building tool to build an interface to the application program.

14. The computer system of claim 13 further comprising:

a visual indicia, for a transaction contained in the application program, which navigates via the link to an entry point of the source code corresponding to transaction.

15. The computer system of claim 14 further comprising:

a visual indicia, displayed with the transaction contained in the application program, which navigates to documentation stored in the repository corresponding to the transaction.

16. The computer system of claim 15 further comprising:

a selected transaction;

a display of a data structure corresponding to the selected transaction;

a selected data structure;

computed input and output fields for the selected data structure; and

an overlay of the fields on the selected data structure.

17. The computer system of claim 16 further comprising:

an editor for editing the computed fields; and

an analyzer for analyzing the selections and editions to determine if an error exists.

18. A computer system for identifying an interface of an application program comprising the interface and source code, said computer system comprising:

a parser for parsing the source code of the application program to identify meta information;

storage for storing in a repository the meta information and a link pointing to an original location of the meta information within the application program;

a query of the repository to determine which source files and which interfaces comprise the application program;

a display of a visual indicia, for a transaction contained in the application program, which navigates via the link to an entry point of the source code corresponding to transaction;

a display of a visual indicia, together with the transaction contained in the application program, which navigates to documentation stored in the repository corresponding to the transaction;

a selected transaction;

a display of a data structure corresponding to the selected transaction;

a selected data structure;

computed input and output fields for the selected data structure;

an overlay of the fields on the selected data structure;

an editor for editing the computed fields;

an analyzer for analyzing the selections and editions to determine if an error exists.

a new source file containing the interfaces which comprise the application program;

storage for storing the new source file and a link pointing to a location of the new source file in the repository; and

a meta language document containing a description of the application program interfaces to enable a connector building tool to build an interface to the application program.